



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
-----------------	-------------	----------------------	---------------------	------------------

10/021,080

12/19/2001

Michael Best

11922-US

8725

23553

7590

06/26/2006

MARKS & CLERK

P.O. BOX 957

STATION B

OTTAWA, ON K1P 5S7

CANADA

EXAMINER

MITCHELL, JASON D

ART UNIT

PAPER NUMBER

2193

DATE MAILED: 06/26/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

10/021,080

Applicant(s)

BEST ET AL.

Examiner

Jason Mitchell

Art Unit

2193

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 30 March 2006.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-9, 12-19 and 21-24 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-9, 12-19 and 21-24 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
 - ☐ Certified copies of the priority documents have been received in Application No. _____.
 - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

This action is in response to a request for continued examination filed on 3/30/06.

At Applicants request claims 1-9, 12 and 22-23 have been amended. Claim 11 has been canceled. Claims 1-9, 12-20 and 22-24 are pending in this application.

Applicant's arguments filed on 3/30/06 have been considered but are moot in view of the new ground(s) of rejection.

Claim Rejections - 35 USC § 103

Claims 1-9, 12-20 and 22-24 rejected under 35 U.S.C. 103(a) as being unpatentable over “The Benefits of CORBA-Based Network Management” by Haggerty and Seetharaman (Haggerty) in view of US 6,898,792 to Burke (Burke).

Regarding Claims 1 and 12: Haggerty discloses a run-time expandable network management and service provisioning framework for use in a network management and service provisioning system.

“an open standards-based, extensible, and distributed network management system using CORBA” (pg. 73, col. 1, paragraph 1)

The framework comprising:

A plug-in registry configured to load and register at run-time at least one enabling-technology plug-in which brokers access to network management and service provisioning enabling technologies.

“The topology objects represent the network and contain information pertaining to addressing, type, uniqueness, resources and status. ... The topology objects are created through OpenView Map additions to the MOM or by auto discovery.” (Pg.

76, col. 1, paragraph 1); "A management system for this network might contain ... an object for each interface, objects modeling connectivity between interfaces, objects for each connection in the network" (pg. 77, col. 1, paragraph 2)

The citations above indicate two things. First the plug-ins (topology objects) are added to the registry (MOM) at run-time ('*OpenView Map additions*' and '*auto discovery*' see ch.1 pp. 2-3 of the OpenView Reference). And second, that these topology objects represent, or broker access to, network management and service provisioning enabling technologies ('*interface*', '*connectivity between interfaces*' and '*each connection*'). The fact that the topology objects represent connections and interfaces places them well within the scope of the claim language.

Haggerty also discloses a directive parser configured to process, at run-time, at least one self-contained managed data network entity specification file including directives.

"The topology objects are created through ... auto discovery." (Pg. 76, col. 1, paragraph 1)

The OpenView auto discovery process, which is preformed at run-time, includes a parsing step and necessarily a parser for parsing entity specifications (see ch.1 pp. 2-3 of the HP OpenView reference).

Haggerty further teaches an implementation of a single managed entity object class, the single managed entity object class being derivable via type derivation into a derivation hierarchy of managed data network object types based on run-time parsed entity derivation directives.

"All objects in the model derive from one base object called a Managed Object. The Managed Object supports an IDL interface that defines base attributes and operations for all objects. ... This abstract model supports all possible network entities. ... In most cases the derived objects do not require much implementation since higher level objects implement most of their properties and functions. In this scenario, adding support for new equipment requires only creating a new object definition, which fits into the model" (paragraph bridging pp. 76-77)

Note that the above citation discloses a 'base' managed entity object class ('Managed Object') from which the individual managed entity objects are derived ('base ... for all objects'; 'derived objects'). Further note that Haggerty's Fig. 4 discloses a hierarchy of managed data network object types. Note even further that these managed entity object class are instantiated using the 'OpenView auto discovery' feature discussed above and thus are 'based on the run-time parsed entity derivation directives'.

Haggerty also discloses a message interpreter processing, at run-time, messages received from at least one network management and service provisioning software application, the message including a software application directive employed to invoke at least one method of a corresponding managed data network object instance of a derived managed data network object type to access a corresponding field installed managed data network entity via the enabling technology plug-in;

"ProSphere includes a client application that integrates with OpenView. This application displays the ProSphere topology within the OpenView graphical interface. Additionally, ProSphere user interfaces can be launched for OpenView windows" (pg. 78, col. 2, par. 2)

First it is noted that the cited 'client application that integrates with OpenView' clearly anticipates a message interpreter processing at run-time messages received from a network management application (e.g. integration). Further, looking to col. 1, par. 1 of

Art Unit: 2193

the same page it is just as clear that the *“ProSphere user interfaces ... interact with the [managed] objects”*.

Still further, Haggerty discloses a separation being achieved between managed data network entities, enabling technologies and software applications, the separation enabling independent development, maintenance and trouble shooting of network management and service provisioning deployments minimizing the need to re-code and re-compile framework code in support of new managed entity object types.

“The ProSphere user interfaces use the compiled stubs from IDL to interact with the objects. This decouples the user interface code completely from the object implementation” (pg. 78, col. 1 par. 1).

Clearly this discloses a separation between the software applications and the objects (network entities and enabling technologies) further, Fig. 4 clearly shows a separation between network entities (managed elements) and enabling technologies (communication). This separation clearly stems from Haggerty's desire to produce *“a network management architecture ... capable of handling new features and changing with networking technology without requiring significant redesign and coding”* (see pg. 76, col. 2, par. 2).

Burke teaches a run-time loadable lexical analyzer stub, associated with a specific management object.

“In a preferred embodiment, encapsulator objects are created by a parser that parses foreign notation. ... the parser accepts foreign object definition information expressed in foreign notation and instantiates one or more encapsulator objects for each rule ... in a grammar that corresponds to the syntactic structure defined in the foreign notation.” (Col. 4, lines 6-17).

Art Unit: 2193

Note that the 'encapsulator object' *"reflects the syntactic structure of the foreign object definition"* and thus must include lexical analysis of directives specified in a 'foreign notation', and that these directives are associated with a managed object (e.g. a self-contained managed data network entity specification).

Further Burke teaches a generic lexical analyzer run-time augmented with the lexical analyzer stub associated with the registered plug-in, the augmented lexical analyzer processing, at run time, an enabling-technology-specific use directive parsed from the managed data network entity specification file;

"The dynamic CORBA/CMIS gateway (203) is a generic application that can accept a variety of requests ... such as CMIS operations ... without compile-time knowledge of the GDMO/ANS.1 definitions for the CMIS operations requested. By using CORBA DSI, the generic gateway can accept invocations ... and parse those invocations using the GDMO/ASN.1 repository. ... In this embodiment, upon receipt of a request using DSI for a CMIS operation from a CORBA object, the dynamic gateway invokes ... a reference to an encapsulator object corresponding to a managed object class template specified by the name" (col. 6, lines 7-26).

Note that the CORBA DSI (Dynamic Interface Skeleton) with assistance from the encapsulator object (specific lexical analyzer) parses and analyzes an 'invocation request' for a 'CMIS operation' (e.g. a use directive). And further note that this happens at run time (*'without compile-time knowledge'*).

It would have been obvious to a person of ordinary skill in the art at the time of the invention to incorporate Burke's specific and generic lexical analyzers in the process of loading Haggerty's 'Topology objects' because then *"The gateway can ... be an off-the-shelf application adaptable through the repository to [a] wide variety of circumstances"* (see Burke col. 6, lines 52-54).

Regarding Claim 2: The rejection of claim 1 is incorporated; further Haggerty discloses the single managed entity object class is an abstract managed object class (pg. 76, col. 2, par. 4 'The ProShpere architecture defines an abstract set of CORBA objects').

Regarding Claim 3: The rejection of claim 1 is incorporated; further Haggerty discloses a derived managed data network object type in the derivation hierarchy includes the specification of at least one attribute (pg. 77, col. 1, par. 1 'the derived objects ... implement most of their properties and functions').

Regarding Claim 4: The rejection of claim 1 is incorporated; further Burke teaches the at least one self-contained managed data network entity specification file includes a human readable file (col. 3, lines 27-29 'Encapsulator objects can include all information contained in GDMO/ASN.1 specification').

Regarding Claim 5: The rejection of claim 4 is incorporated, further Burke teaches the human readable file is an attribute file holding attributes corresponding to a single managed entity object type derivable at runtime in one of a direct and an indirect manner from the single managed entity object class (col. 4, lines 6-17 'the parser accepts foreign object definition information expressed in foreign notation and instantiates one or more encapsulator objects for each rule ... in a grammar that corresponds to the syntactic structure defined in the foreign notation.') Note that Burke also teaches that GDMO/ASN.1 defines attributes (col. 2, lines 33-41 "among other object information, ... transfer syntaxes which can be used to invoke DDMO objects")

Regarding Claim 6: The rejection of claim 3 is incorporated; further Burke teaches the entity derivation directive includes an attribute specification (col. 2, lines 33-41 "among

other object information, ... transfer syntaxes which can be used to invoke DDMO objects”).

Regarding Claim 7: The rejection of claim 6 is incorporated; further while Haggerty does not explicitly disclose the attribute specification further specifies managed data network object type inheritance, he does disclose that the object being derived from the specification allows for inheritance (pg. 77, col. 1, par. 1 ‘higher level objects implement most of their properties and functions’). It would therefore have been obvious to a person of ordinary skill in the art at the time of the invention to further specify managed entity object type inheritance.

Regarding Claim 8: The rejection of claim 1 is incorporated; further Haggerty discloses the network management and service provisioning enabling technologies include support for at least one of a persistence method and a persistence entity (pg. 76, col. 1, par. 2 ‘The topology objects ... contain information pertaining to addressing, type, uniqueness, resources, and status’).

Regarding Claim 9: The rejection of claim 1 is incorporated; further Burke discloses the use directive further specifies a command sequence to be followed in using a specific registered enabling technology (col. 2, lines 33-41 “among other object information, ... transfer syntaxes which can be used to invoke DDMO objects”).

Regarding Claim 13: The rejection of claim 12 is incorporated; further Haggerty discloses processing the at least one message received by the framework, the method comprises a further step of deriving a containment hierarchy of managed data network type instances corresponding to field installed data network equipment (Fig. 4).

Regarding Claim 14: The rejection of claim 12 is incorporated; further Haggerty discloses registering with the framework at least one plug-in, the method further comprises a step of run-time registering the at least one plug-in (pg. 76, col. 1, par. 2 'The topology objects are created through OpenView Map additions to the MOM').

Regarding Claim 15: The rejection of claim 14 is incorporated; further Haggerty discloses wherein run-time registering the at least one plug-in, the method further comprises a prior step of: selecting the at least one plug-in for registration thereof (pg. 76, col. 1, par. 2 'The topology objects are created through OpenView Map additions to the MOM').

Regarding Claim 16: The rejection of claim 12 is incorporated; further Haggerty discloses a step of: prior to parsing the at least one managed data network entity; run-time loading the at least one managed data network entity specification (pg. 76, col. 1, par. 2 'The topology objects are created through OpenView Map additions to the MOM'). Note that in order to parse a thing that thing must first be loaded; otherwise the parser cannot interact with it.

Regarding Claim 17: The rejection of claim 16 is incorporated; further Haggerty discloses run-time loading the at least one managed data network entity specification, the method further comprises a prior step of: selecting the at least one managed data network entity specification (pg. 76, col. 1, par. 2 'The topology objects are created through OpenView Map additions to the MOM').

While Haggerty does not explicitly disclose selecting the at least one managed data network entity specification, It would have been obvious to a person of ordinary skill in

Art Unit: 2193

the art at the time of the invention to provide a user with the ability to select the at least one managed data network entity specification instead of having to re-define the managed data network entity prior to adding it to the MOM.

Regarding Claim 18: The rejection of claim 12 is incorporated; further Haggerty discloses wherein parsing, the method further comprises a step of: extracting at least one directive from the at least one managed data network entity specification, the at least one managed data network entity specification being associated with at least one managed entity object type (pg. 77, col. 1, par. 1 'higher level objects implement most of their properties and functions').

Haggerty does not explicitly disclose extracting at least one directive from the data network entity specification said directive being associated with at least one managed entity object type, however this association is inherent in his disclosure on pg. 77, col. 1, par. 1 'higher level objects implement most of their properties and functions'. Without a directive associated with at least one managed entity object type, inheritance as disclosed could not be established.

Regarding Claim 19: The rejection of claim 12 is incorporated; further Haggerty discloses wherein deriving the single managed entity object class via type derivation, the method further comprises a step of setting at least one attribute (pg. 77, col. 1, par. 1 'the derived objects ... implement most of their properties and functions').

Regarding Claim 20: The rejection of claim 12 is incorporated; further Haggerty discloses wherein prior to processing the at least one message received by the framework from the at least one software application, the method further comprises a

Art Unit: 2193

step of: registering the at least one software application with the framework (Fig. 2, ProSphere Application Objects').

Regarding Claim 22: the rejection of claim 12 is incorporated; further Haggerty discloses wherein implementing the third directive, the method further comprises a step of: instantiating managed entity object types (pg. 76, col. 1, par. 2 'The topology objects are created through OpenView Map additions to the MOM').

Regarding Claim 23: The rejection of claim 21 is incorporated; further Haggerty discloses wherein implementing the one of the second and third directive the method further comprises a step of: effecting a change in a network state of a managed data transport network in a realm of management (pg. 78, col. 1, par. 1 'The ProSphere user interfaces use the compiled stubs from IDL to interact with the objects').

Regarding Claim 24: The rejection of claim 12 is incorporated; further Haggerty discloses wherein subsequent to processing the at least one message received by the framework; the method further comprises a step of: sending a message to the software application (pg. 78, col. 2, par. 2 'integrates with OpenView').

Conclusion

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Jason Mitchell whose telephone number is (571) 272-

Art Unit: 2193

3728. The examiner can normally be reached on Monday-Thursday and alternate Fridays 7:30-5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kakali Chaki can be reached on (571) 272-3719. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.



Jason Mitchell
6/13/06



KAKALI CHAKI
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100